

## Addendum: Notes about added functionality in the new DTC

### Use `dtc -h`

Notice how some of the parameters do change their meaning when used for writing.

DTC for *\*any\** kind of write operation from any kind of image source now decides the best parameters to use (except for precomp settings, since those change during runtime) – and there are lots of them now...

### Caveat:

- Any side to be written must be synced to side 0 index, there is no write splice/track gap detection.

- Weak bits may or may not work, depends on the source drive

- NFA does not work

NFA must be detected and generated with a high frequency, weak bits must be properly detected unless you want to rely on the goodwill of your drive electronics.

**Using the parameters** it is possible to override most of the settings made by DTC – making any user setting prevents the automatic one.

Incorrectly setting the floppy mode (reversed or not); DTC knows that only sampled data can be reversed, so `-y` gets demoted to `-wy`, see below.

Some other parameters may also be demoted or forced to change if it is absolutely necessary.

**There is a short delay after** (at least on a fast PC and a 64 bit system) after displaying the filename.

This is when loading the entire image happens.

After that, basic image information that could be retrieved directly from the image is displayed, such as image type and geometry.

Now there is a longer pause happens, if the image loaded is sampled data, e.g. stream.

How long this wait is very much depends on how fast your PC is, what does the image contain, and whether you use the 64 bit version or the 32 bit one.

The 64 bit version can be several times faster on the same image, so if you can run that, make sure to use it.

Which version is running is always displayed with `dtc` command (`-h` or no command line) Win64 or Win32. You normally want the Win64 one.

**During the wait** all tracks in the image goes through entropy analysis.

Based on the results of that, various filters, user settings (or the lack of them!) a quite complex logic, heuristics decides on each write setting to be used to at least have a chance to succeed ;)

DTC will set all of the write parameters (except for precomp) based on the outcome of the various stages.

Once the analysis complete DTC may issue warnings (operation continues) or errors (write does not commence).

The final parameter settings get displayed the symbolic names, their command equivalent and the value set.

If the specific parameter has already been set by the user and the user setting has been overridden by DTC the user value is in brackets.

E.g. `0[3]` means the user set the parameter to 3, but DTC forced it to 0. When this happens it cannot be overridden at all; there is a very good reason for it, such as physical limits found.

**The summary** of the side statistics also get displayed.

Each side is treated in isolation from the other one, so parameters found for one do not apply to the other. Note, that some of the settings cannot be individually changed by the user (intentionally), but can be by DTC.

**Td: Track distance found.** If this is 1, but you want to make sure it's 2, just use `-k2`, see below.

If DTC finds that `td` is 2 for any side, that side can be safely written with 80 or 40 track drives without loss of data.

If `td` is 1, there are tracks that can only be written with a 80 track drive.

(40 as in 40+, 80 as in 70+, ie 48 tpi, 96/135 tpi, etc)

Note, that it is perfectly legit to have `td: 1` for e.g. a C64 disk. Apart from the cases when filtering cannot safely eliminate all offending tracks, this happens as well when you analyse a XEMAG or other so-called "Fat Track" -protected disk.

**Data: the number of tracks containing data.** These tracks may be anywhere on the side, 40 tracks is not necessarily the first 40 track of the image, but usually is.

**Unformatted: unformatted tracks**, as per entropy and filtering.

The original number of track is the number of tracks as displayed for the image.

The number here is the formatted tracks after entropy analysis and filtering.

If the filtering (or user by using `-k2`) did not remove any tracks, you will see a single value, like 35 which is always after entropy analysis.

If filtering did remove any track the number of tracks after entropy check is in brackets, e.g. `35[68]` means that entropy analysis found 68 tracks with any kind of possibly legit content, and later filters reduced this number to 35.

**Unformatted:** the number of unformatted tracks after entropy analysis and after filtering, see Data.

NFR: tracks that contain no flux reversal at all, see Data. These tracks get demoted to unformatted under specific circumstances; if the side does not contain any legit data track at all. This makes it possible to completely ignore unformatted sides if it is a viable option.

**New write parameters** (notice, some of them have different meaning for read)

**-wy:** Write side 1 to side 0, side 1 becomes unformatted.

The unformatted side is by design specifically to prevent creating fake combinations of disk sides.

This feature can be used to write any kind of floppy disk content on a single sided drive when the direction of the data does not change.

Typically, preprocessed images, like say a Spectrum+3 IPF.

Those 3" drives are single sided, and the images are made by manually flipping the disk.

Writing is obviously the same process, since there is physically no side 1 head.

Write side 0, limit writing to side 0 (-g0), flip the disk in the drive, then add -wy (this automatically uses -g0 unless user selected another value)

**-y: Same as -wy**, but side 1 is transposed to side 0, ie the written data is reversed.

This is useful for data that is NOT pre-processed, such as stream files, where the data was sampled in a single pass for floppy disks (CBM, A8, BBC? Etc)

This automatically enforces -wy

**Trying to flip a disk** in any way when the resulting image would be unformatted (ie side 1 was found to be unformatted during analysis) is not possible.

### **-g: Enforce writing of a side.**

DTC selects the sides to be written automatically; if only one side has content, that side, if both then both.

While this can be a lot faster, and is what is absolutely necessary to have the same disk as the original for disks where both sides were generally used (ST, Amiga...) you may want to enforce -g2 should the program ever decides to check the side that should be unformatted. Usually they don't though, since many of them ended up on recycled disks.

### **-k / -ks**

Default is -k2

-k<step> is the preferred image type of the analysis result - you prefer to have a 40/80 track image if at all possible.

-ks<step> is for enforcing the image type. You do not and should not use e.g. -k2 and -ks2 at the same time. Use only the one you need.

Any -ks command disables crosstalk filtering.

The image analysis output has also been updated to show exactly which -k option was selected and could be used again to reproduce the results.

-k1 allows crosstalk filtering, except for the crosstalk elimination phase as specified at the description of command -k2.

-k2 runs the crosstalk detector and if it finds crosstalk on a side at all, it wipes all potential crosstalk tracks, unless the SNR of the data content of the track to be wiped is the same as the neighbouring tracks. In that case the track will remain untouched.

-ks1 will keep all tracks of the image, regardless of analysis saying otherwise.

-ks2 will wipe every odd track blindly.

Entropy analysis and other filters are not affected, so what those find may still affect the image to be written.

### **-wg: Enable side noise filter** for any of the sides.

The side is actually a bitmask, b0 is side 0, b1 is side 1, ie 0: no filter, 1: filter side 0, 2: filter side 1, 3: filter both sides

It is very unlikely that a side contains various short bursts of random small amount of data by design, this filter eliminates a side if only such data content is found for all tracks.

**-wk: Enable crosstalk filter**, see -wg for defining the sides affected

Crosstalk happens when 80 track drives are used to sample disks intended for 40 track systems; there is often crosstalk on every odd track (1, 3, 5...).

DTC tries to remove these tracks from the image, but the results depend on the drive used for sampling and whether possibly legit data can be found on a track that is potentially just crosstalk.

The approach is to not remove anything unless it is 100% certain that a track only contains crosstalk, e.g. it won't ruin your Amiga disk image, even though it has no idea of what kind of image you are using at all.

The crosstalk filter is automatically activated based on various properties of the entire disk side for each side separately.

If it was activated at all (on either side), changing the write bias to any other value than 0 is the clue, e.g. "bias -wb: 1" displayed as a derived parameter. When write bias changes, so does write erasure as well; using a bias only makes sense if the entire side is wiped first.

DTC will just use normal writing if no crosstalk was detected at all.

**There is a trick to see the result** of the image analysis without writing or having a KF attached.

Dtc -w -m1 -fimage/\* -i -fimage/<your real image> <your parameters>

This would try to write into the stream image in the first parameter, which will only fail once the actual writing is starting.

\* should be any stream file, usually \* is enough, but if you try what happens with adf, g64, ipf make sure there is a stream file accessible as well, otherwise DTC finds out that the device is invalid for writing before the analysis. This is a diagnostic feature.